NBSIR 82-2591

Adaptive Algorithm for the Control of A Building Air Handling Unit

U.S. DEPARTMENT OF COMMERCE
National Bureau of Standards
National Engineering Laboratory
Center for Building Technology
Building Equipment Division
Washington, DC 20234

Service Networks Planning Department
Bell Laboratories
Holmdel, NJ 07733

October 1982

Sponsored by:
Office of Buildings and Community Systems
U.S. Department of Energy
U.S. Navy Civil Engineering Laboratory
U.S. Department of Defense

NBSIR 82-2591

ADAPTIVE ALGORITHM FOR THE CONTROL OF A BUILDING AIR HANDLING UNIT

Cheol Park

U.S. DEPARTMENT OF COMMERCE
National Bureau of Standards
National Engineering Laboratory
Center for Building Technology
Building Equipment Division
Washington, DC 20234

Alexander J. David*

Service Networks Planning Department
Bell Laboratories
Holmdel, NJ 07733

October 1982

Sponsored by:
Office of Buildings and Community Systems
U.S. Department of Energy
U.S. Navy Civil Engineering Laboratory
U.S. Department of Defense

U.S. DEPARTMENT OF COMMERCE, Malcolm Baldrige, *Secretary*
NATIONAL BUREAU OF STANDARDS, Ernest Ambler, *Director*

ABSTRACT

The use of adaptive control algorithms was studied for microprocessor driven direct digital control of elementary heating and cooling subsystems. An algorithm was designed for digital regulation of a linear, time-invariant first-order system with a system dead time. A recursive least squares algorithm was used to estimate, on-line, the parameters of the time-invariant linear system. The parameter estimates were then used to calculate the feedback gains of a Proportional plus Integral (PI) controller.

Through computer simulations, the adaptive-parameter PI-controller was compared with a constant-parameter PI-controller. On the basis of favorable simulation results, the adaptive algorithm was implemented for direct digital control of an air handling unit in a laboratory building at the National Bureau of Standards, Gaithersburg, Maryland. The convergence of the parameter estimates and the step response proved to be satisfactory provided the system was operating in a linear or weakly non-linear region, and was in steady or quasi-steady state. By selecting a proper scale factor, improved performance may be obtained when system characteristics vary.

Key words: adaptive control; air handling unit; direct digital control; energy management and control systems; HVAC system control; parameter estimator; PI-controller; recursive least squares algorithm; self-tuning control algorithm

---

* Dr. David's contribution to this research effort was carried out while he was employed by NBS from April 1980 to May 1981.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

d	dead time
e	error (= r - y)
e_R	operating temperature range
K	process gain
K_p	proportional gain of controller
$\underline{P}$	covariance matrix
p	pressure
r	reference value (set point value)
t	time
t_ℓ	accumulated leftover time
T	sampling period of process simulation
T'	sampling period of controller
T_I	integral time constant of controller
T_{OA}	outside air temperature
T_R	return air temperature
S_F	scale factor
u	input to process model
u_c	controller output
u_R	limiting value of u_c
u_v	valve output
V	loss function
w	disturbance
x_i, $i = 1, 2$	state variables of process
y	process output

α	inverse of time constant of process
β	ratio of T_{OA} to T_R
δ	term used in z-transform
Δp	pressure difference
Δt_a	actuator operating time
Δu_c	controller output difference
ε	residual in estimation
$\underline{\theta}$	parameter estimate vector
λ	"forgetting factor"
$\underline{\phi}$	regressor vector

1. INTRODUCTION

A building system requires an integrated system of controllers to provide the best possible level of comfort with a minimum level of energy consumption. One well-known type of controller is the Proportional plus Integral plus Derivative (PID) controller. Conventionally, PID controllers are tuned manually to achieve accurate performance. Unfortunately, this manual process is not a simple task because of many factors to be accounted for, and moreover, a controller properly adjusted for one season is not likely to perform properly at a different time of year. This problem suggests the use of modern estimation and control algorithms for performing the tuning process in real-time.

The benefits of using microprocessors for tuning controllers have been recognized in many industries [1-4]. The parameters of a controller can be adaptively updated based on estimates of the parameters of the system to be controlled [5-7] using the methods of modern control theory. There are two recognized classes of adaptive control methods--the model reference method [8] and the self-tuning regulator [9].

Kurz, Isermann, and Schumann [10] have compared various parameter-adaptive control algorithms. They indicate that parameter-adaptive control can be used for tuning of digital control algorithms, for digital adaptive control of slowly time-varying processes, and for digital adaptive control of weakly non-linear and partially unstable processes.

For our purposes, the self-tuning regulator approach seemed most appropriate. The study was begun by deriving a model of a simple heating, ventilation and air conditioning (HVAC) system and using this model to simulate the performance of a control system consisting of a recursive least squares algorithm for parameter identification and an adaptive PI-algorithm. Based upon the simulation results, an algorithm was developed for direct digital control. In laboratory experiments, the algorithm was employed in microcomputer-based direct digital control of an air handling unit.

2. MATHEMATICAL MODEL

For simulation purposes, we modeled an air handling unit as shown in figure 1. Outdoor air enters the system through the opening of dampers and is mixed with return air. The mixed air passes through an air filter, a heating coil, and a cooling coil. A fan supplies the conditioned air to the demand zone [11]. In this simple system, the mass flow rate of hot steam is controlled by a proportional valve, which is governed by a controller. The controller operates to minimize the difference between the temperature measured by a sensor in the supply duct and the set point value. The flow rate of chilled water is assumed to remain constant. For simplicity, feedback by a room thermostat is not considered. The problem addressed is to find algorithms for tuning conventional digital controllers by providing appropriate system parameter values. Thus, we replace the conventional controller with an adaptive digital controller implemented via a microcomputer.

Following the self-tuning regulator methodology, we depict the interrelationships among the elements of the control system in figure 2. The control system consists of a process, a valve, a controller, and an estimator. The process produces the output $\tilde{y}$ using the change of valve output Δu_V as an input. The process output

then combines with outdoor and return air disturbances w resulting in y. The difference between the resultant output and the set point value (reference value) r yields an error e. When parameters of the controller are fixed, the controller output u_c depends on the error signal. But if the controller is adaptive, it depends not only on the error e but also on the new estimates of $\underline{\theta}$ of the system parameters as obtained by the estimator. The estimator requires the current input value u_c and output value y as its inputs as well as prior values of u_c and y in order to update the parameters. Details are given below.

2.1 Process Simulation

We consider the heating and cooling coils as the process, assuming that the chilled water going through the cooling coil has constant temperature and flow rate. Under this assumption, we simulate the heating coil as the process, treating it as a continuous linear device, with Laplace transform

$$G_H(s) = \frac{Ke^{-ds}}{s + \alpha} \tag{1}$$

where K, d, and α are the gain, dead time (transport lag), and inverse time constant of the system, respectively. The time-invariant values of K, d, and α are determined prior to simulation.

Assuming that the control signal is changed discretely every T' = nT time units, where T is the simulation sampling interval, the system model must incorporate a zero order hold in cascade with the coil. The transfer function of the sampler is

$$G_s(s) = \frac{1 - e^{-nTs}}{s} \tag{2}$$

From equations (1) and (2), the transfer function of the process with the sample and hold yields in the frequency domain

$$H(s) = G_H(s)G_s(s)$$

or

$$H(s) = \frac{(1 - e^{-nTs})(Ke^{-ds})}{s(s + \alpha)} \tag{3}$$

Equation (3) can be written as

$$H(s) = [e^{-ds} - e^{-(nT + d)s}](K)(\frac{1}{s})(\frac{\frac{1}{s}}{1 + \alpha \cdot \frac{1}{s}}) \tag{4}$$

Using the block diagram convention [12], we depict equation (4) in figure 3. where x_1 and x_2 are state variables. From this block diagram, we obtain a set of state equations based on the dynamic linear network theory. These equations in the time domain are

$$y = x_1$$

$$\dot{x}_1 = x_2 - \alpha x_1$$

$$\dot{x}_2 = Ku$$

where

$$u(t) = u_v(t - d) - u_v(t - d - nT)$$

In a matrix form, we obtain

$$\underline{\dot{x}} = \underline{A}\,\underline{x} + \underline{B}\,u \tag{5}$$

$$y = \underline{C}\,\underline{x}$$

where

$$\underline{\dot{x}} = [\dot{x}_1,\ \dot{x}_2]^T,\ \underline{x} = [x_1,\ x_2]^T$$

$$\underline{A} = \begin{bmatrix} -\alpha & 1 \\ 0 & 0 \end{bmatrix},\ \underline{B} = \begin{bmatrix} 0 \\ K \end{bmatrix} \tag{6}$$

$$\underline{C} = [1\ \ 0]$$

In equations (5) and (6), u and y are scalar quantities since the system is a single input single output system. Given a dynamic linear system expressed by equations (5) and (6) with the initial state $\underline{x}(t_o)$ and the input u(t) for $t \geq t_o$, we seek the output y(t) for $t \geq t_o$. Since the input u(t) is expressed in the form of sampled data, both $\underline{x}(t)$ and y(t) are determined only for discrete multiples of the simulation sampling interval T.

When u(t) is a piecewise-linear function and $\underline{A}$ is time-invariant we obtain an approximate solution [13] for $\underline{x}(t)$ as

$$\underline{x}(kT) = e^{\underline{A}\,T}\,\underline{x}[(k-1)T] + e^{\underline{A}\,T}\,\underline{B}\,\frac{T}{2}\,u[(k-1)T] + \underline{B}\,\frac{T}{2}\,u(kT) \tag{7}$$

where k is an integer. For the $\underline{A}$ matrix of equation (6), the matrix exponential $e^{\underline{A}T}$ is expressed in closed form as follows [14]:

$$e^{\underline{A}T} = \begin{bmatrix} e^{-\alpha T} & \frac{1 - e^{-\alpha T}}{\alpha} \\ 0 & 1 \end{bmatrix} \tag{8}$$

Assuming the dead time d to be equal to mT, where m is an integer, we can express the input function u in terms of the valve output u_v.

$$u(kT) = u_v[(k - m)T] - u_v[(k - m - n)T] \tag{9}$$

The state equation for simulating the linear process is thus:

$$\begin{bmatrix} x_1(kT) \\ x_2(kT) \end{bmatrix} = \begin{bmatrix} e^{-\alpha T} & \frac{1 - e^{-\alpha T}}{\alpha} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x_1[(k - 1)T] \\ x_2[(k - 1)T] \end{bmatrix} +$$

$$\begin{bmatrix} e^{-\alpha T} & \frac{1 - e^{-\alpha T}}{\alpha} \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ \frac{KT}{2} \end{bmatrix} u[(k - 1)T] + \begin{bmatrix} 0 \\ \frac{KT}{2} \end{bmatrix} u(kT) \tag{10}$$

The process output y is given by

$$y(kT) = x_1(kT) \tag{11}$$

Equations (9) through (11) are used for the linear process simulation with the piecewise linear input u which is, in turn, the change of valve opening. For given T, the parameters describing the process--α, K, and m--are time-invariant.

2.2 Valve Simulation

The valve works as a final control element. For the simulation of the valve operation, we assume a non-linear valve opening with respect to the controller output signal. The response of the valve is shown in figure 4. A possible hysterisis of the valve action is ignored.

2.3 Controller Simulation

As a simple approach, we take a PI-controller. An ideal PI-controller [15] has a transfer function for a continuous system represented by

$$\frac{U(s)}{E(s)} = K_p\left(1 + \frac{1}{T_I s}\right) \tag{12}$$

where U(s) and E(s) are Laplace transforms of the controller output signal u_c, and the error signal e, respectively. K_p and T_I are the proportional gain of the controller and the integral time constant, respectively.

Because we deal with sampled data, we perform the z-transformation on equation (12) to obtain in the first order approximation

$$\frac{U(z)}{E(z)} = K_p + \frac{K_p}{T_I}\,\frac{T'z}{z-1} \tag{13}$$

or

$$U(z) - z^{-1}\,U(z) = K_p E(z) - z^{-1} K_p E(z) + \frac{K_p T'}{T_I} E(z) \tag{14}$$

where T' is the controller sampling interval.

The difference equation corresponding to equation (14) is thus

$$u_c(kT') = u_c[(k-1)T'] + K_p\left(1 + \frac{T'}{T_I}\right)e(kT) - K_p e[(k-1)T'] \tag{15}$$

Equation (15) represents an algorithm for the discrete PI-controller with the parameters, K_p and T_I, to be determined a priori.

The choice of appropriate values of K_p and T_I is commonly [16] made in accordance with the Ziegler-Nichols guidelines [17]. For a single input single output system in which the process is as assumed in equation (1), the Ziegler-Nichols criteria recommend:

$$K_p = \frac{C_p}{Kd} \qquad \text{and} \qquad T_I = C_I d, \tag{16}$$

where

$$C_p = 0.9, \qquad \text{and} \qquad C_I = 3.3.$$

The values of K_p and T_I are constant for a non-adaptive control system. For an adaptive controller, K_p and T_I are updated through estimates of K and d, as explained in the following development.

The z-transform corresponding to equation (3) [18] is:

$$H(z) = \frac{Y(z)}{U(z)} = \frac{b_1 z^{-\ell} + b_2 z^{-\ell-1}}{1 + a_1 z^{-1}} \tag{17}$$

where ℓ is the next larger integer than $\left[\frac{d}{T'}\right]$. (The bracket notation indicates taking the integer part).

Thus $d = \ell T' - \delta T'$ for some δ such that $0 \leq \delta < 1$.

The coefficients in equation (17) are

$$a_1 = -e^{-\alpha T'} \tag{18}$$

$$b_1 = \frac{K}{\alpha}(1 - e^{-\alpha\delta T'}) \tag{19}$$

$$b_2 = \frac{K}{\alpha}(e^{-\alpha\delta T'} - e^{-\alpha T'}) \tag{20}$$

In the time domain, the process output can be represented from equation (17) as

$$y(kT') = -a_1\, y[(k - 1)T'] + b_1 u[(k - \ell)T'] + b_2 u[(k - \ell - 1)T']. \tag{21}$$

We can thus write

$$y(kT') = \underline{\theta}^T\, \underline{\phi} \tag{22}$$

where

$$\underline{\theta} = [\theta_1, \theta_2, \theta_3]^T = [a_1, b_1, b_2]^T \tag{23}$$

$$\underline{\phi} = [\phi_1, \phi_2, \phi_3]^T$$

$$= \left\{-y[(k - 1)T'], u[(k - \ell)T'], u[(k - \ell - 1)T']\right\}^T \tag{24}$$

Since the parameters, a_1, b_1, and b_2, are to be determined adaptively, α, δ, and K are time-dependent. The expressions of α, δ, and K in terms of θ_1, θ_2, and θ_3 are obtained from equations (18), (19), (20), and (23).

$$\alpha = -\frac{\ell n(-\theta_1)}{T'} \tag{25}$$

$$K = \frac{\alpha(\theta_2 + \theta_3)}{1 + \theta_1} \tag{26}$$

$$\delta = -\frac{1}{\alpha T'} \ell n(\frac{\alpha\theta 3}{K} - \theta_1) \tag{27}$$

Using the integer, ℓ, which is given by

$$\ell = \left[\frac{d}{T'}\right] + 1,$$

we write equations (16) in terms of ℓ and δ as:

$$K_p = \frac{C_p}{K\left\{\left[\frac{d}{T'}\right] + 1 - \delta\right\} \quad T'} \tag{28}$$

$$T_I = C_I \left\{\left[\frac{d}{T'}\right] + 1 - \delta\right\} T' \tag{29}$$

The controller outputs are determined by equation (15) with equations (25) through (29) when the parameters θ_1, θ_2, and θ_3 are obtained by the estimation routine.

2.4 Parameter Estimator

To obtain the needed estimates of these system parameters, we use the well-known method of recursive least squares estimation [5, 6, 9]. The recursive least

squares algorithm, under quite general conditions, minimizes the loss function given by:

$$V = \sum_{j=1}^{t} \lambda^{t-j} \varepsilon^2(j) \quad , \qquad 0 < \lambda \leq 1 \tag{30}$$

where ε is the residual between the estimates of the system parameters of time j and their true values. The use of $\lambda < 1$ gives lower weight to less recent data. To implement the recursive least squares algorithm, we need to use a different form of process model, namely the auto-regressive moving average (ARMA) time series model:

$$y(t) + a_1 y(t - 1) + \ldots + a_m y(t - m) = b_1 u(t - d) + b_2 u(t - d - 1) + \ldots$$
$$+ b_n u(t - d - n + 1), \tag{31}$$

where y(t) is the process output at time t and u(t) is the process input at time t.

Note that m and n in equation (31) are integers, and the transport delay (dead time) is d. In more compact notation, a regressor vector and a parameter vector defined as:

$$\underline{\theta} \equiv [\theta_1, \theta_2, \ldots \theta_{m+n}]^T \tag{32}$$

$$= [a_1, a_2, \ldots a_m, b_1, b_2, \ldots, b_n]^T$$

$$\underline{\phi} \equiv [\phi_1, \phi_2, \ldots \phi_{m+n}]^T$$

$$= [-y(t-1), -y(t-2), \ldots, -y(t-m), u(t-d), u(t-d-1), \ldots, u(t-d-n+1)]^T, \tag{33}$$

allow the model given by equation (31) to be written as (Recall equations (23) to (27)):

$$y(t) = \underline{\theta}^T \underline{\phi}. \tag{34}$$

The recursive least squares parameter estimation algorithm [9] is given by

$$\underline{\theta}(t+1) = \underline{\theta}(t) + \underline{P}(t+1)\underline{\phi}(t+1)\varepsilon(t+1) \tag{35}$$

where

$$\varepsilon(t+1) = y(t+1) - \underline{\theta}^T(t)\underline{\phi}(t+1) \tag{36}$$

$$\underline{P}(t+1) = [\underline{P}(t) - \underline{P}(t)\underline{\phi}(t)R(t)\underline{\phi}^T(t)\underline{P}(t)]/\lambda \tag{37}$$

$$R(t) = [\lambda + \underline{\phi}^T(t)\underline{P}(t)\underline{\phi}(t)]^{-1}. \tag{38}$$

If λ is equal to one, the magnitude of the trace of the $\underline{P}$ matrix decreases monotonically. If λ is less than one, the trace of $\underline{P}$ increases. The gain of the estimator thus depends partially upon λ, which is determined empirically.

The properties of the $\underline{P}$ covariance matrix as a function of λ are discussed extensively by Åström [9]. He points out possible instability of the $\underline{P}$ matrix and suggests a number of remedies. One of his suggestions is adapted in our

implementation of the recursive least squares algorithm. When the value of $\Sigma \underline{P}(t)\underline{\phi}(t)$ is less than a given small value, the updating of the $\underline{P}(t)$ matrix is stopped.

To obtain a system model in the required form, so that the vectors $\underline{\theta}$ and $\underline{\phi}$ are properly identified, we must return to equation (3). However, the estimation algorithm observes the system inputs and outputs only at the controller sampling interval T'. Thus, the continuous time system to be modelled has Laplace transform:

$$H(s) = \frac{(1 - e^{-T's})(Ke^{-ds})}{s(s + \alpha)} \qquad (39)$$

3. COMPUTER SIMULATION

3.1 Procedure

Using the mathematical model of the process, we performed computer simulation studies, combining both a constant PI-controller and an adaptive PI-controller. In our simulations of the adaptive controller we assumed initial conditions for the covariance matrix $\underline{P}$ and the parameter vector $\underline{\theta}$ as

$$\underline{P}(0) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \text{ and } \underline{\theta}(0) = [-1,0,0]^T. \qquad (40$$

Steady state values were used as initial conditions of the process.

We employed two different forgetting factors λ; one which was less than unity to speed up the convergence and the other factor which was unity to

keep the covariance matrix from instability. After beginning the simulation or setting a new reference value, we used $\lambda = 0.98$ for 30 sampling periods for fast convergence of the parameter estimates. However, since the trace of the covariance matrix grew rapidly when λ was equal to 0.98, we switched to $\lambda = 1.0$, subsequently, which served to stabilize the covariance matrix.

Although it is desirable to take one λ-value which gives stable values of $\underline{P}$, we reinitialized the simulation with new sets of initial conditions every 24 hours.

As seen in equations (25) and (27), $\underline{\theta}$ is one of the arguments of a logarithm function. To ensure positive values of α and K, and to satisfy the boundedness of δ we imposed limits on the parameter estimates to assure that:

$$\theta_1 < 0, \qquad \theta_2 > 0, \qquad \theta_3 > 0.$$

We also examined the values of $\sum_{i,j} P_{i,j}(t)\phi_j(t)$. If the growth rate of the summed quantity of $\underline{P}(t)\underline{\phi}(t)$ was less than and equal to 1.0 percent, we stopped the parameter update. When the growth rate became more than 1.0 percent, we resumed the parameter update.

To initialize the controller, we needed to have estimates of the dead time d, the process time constant $\frac{1}{\alpha}$ and the process gain K. The valve response characteristics were determined prior to simulation as shown in figure 4. As noted above, two sampling intervals were required, one for the process and the other for the controller and estimator. We took one minute interval for controller and estimator sampling and one second interval for process sampling. The controller sampling time should be equal to or less than the

system dead time for good stability. (Recall the assumption on d for equation (17).)

At first, we simulated the process with a constant outdoor air temperature of -9.7°C (14.6°F) for given process time constant, gain, and dead time, and assumed that there was no return air. A step change in setpoint was the only disturbance. From this simulation, we observed the behavior of the parameter estimates and the covariance matrix and the influence of various values for the forgetting factor λ.

Next we considered the outdoor air temperature, as varying minute by minute. We took a typical daily cycle of mean outdoor air temperature T_{OA} [19]. The maximum and minimum air temperatures were taken as 0°C (32°F) and -11.1°C (12°F) respectively.

We considered the impact of a stochastic component in the disturbance measurement by adding in pseudo random numbers v. A general form of the disturbance w can be given simply as:

$$w = (1 - \beta)T_R + \beta T_{OA} + v \tag{41}$$

where β, T_R and T_{OA} are the ratio of outdoor air to supply air, the return air, and outdoor air temperatures, respectively. In our simulations, we fixed β to be 1.0, which means that 100 percent outdoor air was used. The resultant process output was thus the sum of the process output due to controller action and that due to the disturbance w. The difference of the resultant process output from the set point value was the error signal that drove the controller.

The PI controller output u_c was initially set to 0.5 and bounded in the interval of [0,1]. The controller output u_c is the input to the valve. The variation of the valve output is the input to the linear process, while the controller output is the input of the estimator.

3.2 Results and Discussion

The purpose of the computer simulations was to test the validity of the process modeling, to compare the performance of the adaptive control algorithm with the constant PI-algorithm, and to provide a basis for implementation of the algorithm in hardware for direct digital control. We observed the system responses due to setpoint change, outside air variations, and change of process parameters.

Figure 5 shows process outputs both with adaptive K_p and T_I and with non-adaptive K_p and T_I. The initially assumed process here was

$$G_H(s) = \frac{35\ e^{-1.0s}}{s + 0.5} \tag{42}$$

The parameters, K_p and T_I of the constant PI-controller, were obtained using the actual values of K, α, and d of the process. These were $K_p = 0.0147$ and $T_I = 3.3$. The adaptive controller needed only one value, the dead time, prior to simulation. We can use initial values of $\underline{\theta}$ and $\underline{P}$ for all cases if plant characteristics do not change drastically.

We presume that the dead time can be estimated adequately in advance and does not change drastically under operating conditions. Mean outdoor air temperature varied during the simulation period from 0:00 a.m. to 8:00 a.m., and a stochastic component with a standard deviation of 0.14°C (0.25°F) was added to the mean temperature.

As seen in figure 5, the adaptive algorithm took about 20 minutes to settle down due to an initial learning period. As time went on, the parameter estimates tended to the actual values. The constant PI-controller damped more quickly than the adaptive controller initially, but after 120 minutes, the process output of the adaptive controller had approached the reference value of 14.4°C (58°F) while the output of the non-adaptive controller was still oscillatory.

The set point was changed to 17.8°C (64°F) at 180 minutes after start-up. Both outputs converged quite well to the new set point. At the 300 minute mark, the process parameters were changed; α from 0.5 to 0.6 and K from 35.0 to 40.0. This time when the set point was changed from 17.8°C (64°F) to 14.4°C (58°F), both responses were less stable. However, even though the output of the constant PI-controller remained oscillatory for a 120 minute period, the output of the adaptive controller stabilized within 40 minutes. From visual inspection one can see that the adaptive controller, in general, is more effective than the non-adaptive controller. We note that a discrete change in process parameters is quite artificial. It is done here merely to illustrate that the constant controller is very dependent on being correctly tuned.

When the valve operation was in a linear region of its response curve, simulations were reasonably good. But we obtained unpredictable results when the valve response was nonlinear. We observed nonlinear operation during a start-up period and a set point change period because of large disturbances. When this learning period was over, the simulation was in a linear region. For the constant PI-controller, we assumed a linear valve.

In figure 6, we compare the process outputs controlled by the adaptive controller and those controlled by the non-adaptive controller for process dead time of 3.0 minutes. The constant PI-controller was assumed to be badly tuned with $K_p = 0.0009$ and $T_I = 9.9$, which were computed based upon $\alpha = 0.01$ and $K = 200$. Although these estimated values were quite extreme, it was interesting to see how much off-set would occur under such extreme cases. All other conditions were identical with those in figure 5, except for the initial values of K_p and T_I. The adaptive algorithm started with the initial condition given by equation (40). The K_p and T_I of the adaptive PI-controller converged to 0.0086 and 11.75, respectively, at the 180 minute mark, while those of the non-adaptive controller remained as $K_p = 0.0009$ and $T_I = 9.9$.

Until the 360 minute point in figure 6, the outputs of the constant PI-controller had considerable off-set. But after 360 minutes, where the set point was changed from 17.8°C (65°F) to 14.4°C (58°F), the off-set became quite small while the outputs of the adaptive algorithm also had noticeable errors when the process parameters were varied. The adaptive algorithm converges slowly when a change in the process characteristics takes place.

Figures 5 and 6 also illustrate the performance of the adaptive control under changes in set point. The experiments showing performance under dramatic changes in process characteristics show the robustness of the adaptive algorithm: even though such parameter changes can only happen very gradually (e.g., seasonally) the adaptive controller can handle them quite effectively.

4. DIRECT DIGITAL CONTROL EXPERIMENTS

4.1 Procedure

Using the adaptive controller described above for direct digital control we performed experiments on an air handling unit in a general purpose laboratory building at the National Bureau of Standards (NBS), Gaithersburg, Maryland. The control of the air handling unit differed somewhat from the computer simulation model. The valves of the air handling unit were actuated pneumatically. In the simulation model, the valve opening for the heating coil was governed directly by the control signal as if the valve was operated depending directly upon the electric voltage level. The apparatus has been described elsewhere [20,21] including the microprocessor-based controller, the digital-to-pneumatic interface, and the associated software.

The test system with adaptive control is depicted in figure 7. Internal to the controller, the signal u_c is produced based upon the error e and the estimated parameters $\underline{\theta}$. It is stored for d sampling units to produce the controller output

$$\Delta u_c = u_c(t - d) - u_c(t - d - 1).$$

The quantity Δu_c is multiplied by a scale factor S_F to become the operating time Δt_a of the motorized actuator during a sampling period. The actuator regulates the branch pressure p in the pneumatic system, which modulates two steam valves for the pre-heat coils, one chilled water valve, and the dampers for the inlet, return, and exhaust air. To avoid complexity, our experiments were made when the damper opening was in a steady-state condition. Hence dampers are not shown in figure 7.

The difference in branch pressure Δp causes a valve opening change of Δu_v, which could not be monitored. The actual valve opening u_v determines the flow rates of the hot steam, chilled water, or both. The air passing through the coils combines with the disturbances w to become the supply air with temperature y. The supply air temperature is measured at the inlet of the supply air fan, which operates continuously.

The motorized pressure regulator (actuator), valves, and coils comprise a process described in equation (1). Thus the operating time of the actuator is an input to the process. Nonlinearity, changes in gain and time constant, changes in dead time, and hysteresis in the process were all observed. An effective adaptive controller can be expected to compensate for many different changes in the process, provided they do not occur too rapidly.

With experience, it was possible to improve the effectiveness of the control algorithm. Unlike the computer simulation, the direct digital control on the real hardware had a narrow range of operation which did not allow the supply temperature to be raised by more than 24°C (75.2°F), which had been set as a high limit.

Thus, restrictions on the controller output and the parameter estimates were needed such that $u_c \in \left[-u_R, u_R\right]$ and $\theta_1 \in (-1,0)$. The limiting value u_R was computed from the following expression:

$$u_R = K_{p,max}\left(1 + \frac{T'}{T_{I,0}}\right) e_R \qquad (43)$$

where e_R is the operating temperature range, $K_{p,max}$ is the maximum value of K_p, and $T_{I,0}$ is the integral time given by equation (29) with $\delta = 0$. Based upon

previous work [20], we set $K_{p,max}$ to 0.1, and e_R to 10.0°C. Equation (43) was taken to have a similar form of equation (15), but it was a choice of convenience. Furthermore, the coefficient C_p in equation (28) and C_I in equation (29) were set to 0.5 $K_{p,\ max}$ and 1.1, respectively.

The scale factor S_F is an empirical conversion of the change in control signal into the actuator operating time. It proved to be of great importance in our tests. The following relationship was used:

$$\Delta t_a = -S_F \Delta u_c + t_\ell \qquad (44)$$

The constant t_ℓ represents the leftover time in a sampling period after the actuator has operated. This was needed since the motorized actuator could not operate for a time interval of less than 0.1 seconds. The FID operating system [21] was capable of accumulating this leftover time until it was greater than or equal to 0.1 seconds. The constant t_ℓ may be omitted from the r.h.s. of equation (44) for lower values of $|t_\ell/\Delta t_a|$.

In the parameter estimator, the initial conditions of the parameter vector $\underline{\theta}$ were set to $\underline{\theta}(0) = [-0.9,\ 0,\ 0]^T$. The forgetting factor λ was employed in the same manner as in the computer simulation. After 24 hours of operation, it was reinitialized at the value 0.98. A flow chart showing essential parts of the algorithm is given by figure 8. The computer program of the adaptive control portion is appended.

4.2 Results and Discussion

Tests were performed from mid-December 1981 until the end of February 1982. During this period, the outdoor temperature varied from mild to severely cold.

Most of our testing was done to determine the system response to a step change of the set point between 15.0°C (59°F) to 18.3°C (65°F).

For a set point of 18.3°C, figure 9 shows time series of the process output y, the error e, the parameter estimates θ_i, the integral time T_I, the proportional gain K_p, the control signal u_c, and the actuator operating time Δt_a. The adaptive control algorithm was initialized at the beginning of the run. Table 1 describes the operating parameters for the controller for this run.

Figure 9 shows that the parameter estimates converged after 60 samples (i.e., 20 min). The value of K_p remained at its given maximum of 0.1, while T_I varied actively as $\underline{\theta}$ changed. The output y took longer, about 80 minutes to settle at reference value. Consequently, the control signal u_c was large until y became steady. The signals of y and Δt_a are very similar.

Figure 10 represents a typical response associated with set point changes. We maintained reasonably constant process outputs prior to the set point change. Since the scale factor depends on the characteristics of the actuator and its interface with the controller, two different scale factors S_F were applied to improve the system response characteristics. That is, when the set point was changed, the scale factor was changed simultaneously, using $S_F = 30.0$ for $r = 15.0$°C, and $S_F = 20.0$ for $r = 18.3$°C. The scale factors were selected so as to provide a consistent level of excitation in the output signals. The selection was empirical, based on past observations.

The scale factor is very important to the adaptive PI-control, and must be estimated carefully for the best performance.

As shown in figure 10, there were overshoots in y when the reference was changed (step change). As would be expected, the rate of damping and the amount of overshoot are strong functions of S_F. Parameter estimation was quite rapid after disturbances due to step changes. Figure 11 shows an example of good performance due to satisfactory choice of S_F.

When the direct digital control experiments are compared with the computer simulations, one sees reasonably good agreement in output damping characteristics. When the process is stable, as assumed in the computer simulations, the adaptive PI-controller becomes a constant PI-controller once the learning period is over.

5. CONCLUSION

Our testing has shown that a self-tuning PI-controller employing the recursive least squares estimator can be used satisfactorily for control of an air handing unit. When the unit is operating in a linear region, performance is particularly accurate. Even in nonlinear operation regions, performance is generally better than that of a constant parameter controller. Moreover, although the direct digital controller that was implemented had a different interface between the valve and controller than the mathematical model, good agreement was observed between experiment and simulation.

For the purposes of commercial implementation, the most important result of our testing was the observation of the importance of the sacle factor, S_F, which may be interpreted as the product of the component gains in the system. It was seen that the correct choice of the scale factor permits greater efficiency of operation of the controller and parameter estimator. More research is needed to develop a procedure that includes a method for changing the scale factor because the gain, time constant, and dead time of HVAC systems depend on the supply air temperature. These dependencies, along with non-linearities in the system are not included in the system model discussed in this report.

ACKNOWLEDGMENTS

The authors are indebted to Dr. George E. Kelly and Dr. Bent A. Borresen for their helpful suggestions, and Mr. C. Warren Hurley and Mr. William B. May, who provided hardware and software for experiments.

REFERENCES

1. Cegrell, T. and Hedquist, T., "Successful Adaptive Control of Paper Machines," Automatica, Vol. 11, 1975, 53-59.

2. Borrison, U. and Syding, R., "Self-Tuning Control of an Ore Crusher," Automatica, Vol. 12, 1976, 1-7.

3. Dexter, A. L., "Self-Tuning Optimum Start Control of Heating Plant," Automatica, Vol. 17, 1981, 483-492.

4. Farris, D. R. and McDonald, T.E., "Adaptive Optimal Control-An Algorithm for Direct Digital Control," ASHRAE Trans. Vol. 86, Part 1, 1980, 880-893.

5. Åström, K.J., and Eykhoff, P., "System Identification - A Survey," Automatica, Vol. 7, 1971, 123-162.

6. Åström, K. J. and Wittenmark, B., "On Self-Tuning Regulators," Automatica, Vol. 9, 1973, 185-199.

7. Åström, K. J., Introduction to Stochastic Control Theory, Academic Press, New York, 1970.

8. Landau, Y.D., Adaptive Control, Marcell Dekker, New York, New York, 1979.

9. Narendra, K. S. and Monopoli, R. V., Applications of Adaptive Control, Academic Press, New York, 1980, 1-68.

10. Kurz, H., Isermann, R. and Schumann, R., "Experimental Comparison and Application of Various Parameter-Adaptive Control Algorithms," Automatica, Vol. 16, 1980, 117-133.

11. Haines, R. W., Control Systems for Heating, Ventilating and Air Conditioning, Van Nostrand Reinhold, New York, 1971.

12. Kuo, B. C., Automatic Control Systems, Prentice-Hall Englewood Cliffs, New Jersey, 1962.

13. Chua, L. O. and Lin, P., Computer-Aided Analysis of Electronic Circuits, Prentice-Hall, Englewood Cliffs, New Jersey, 1975.

14. Chen, C., Introduction to Linear System Theory, Hold, Reinhard and Winston, 1970.

15. Bibbero, R. J., Microprocessors in Instruments and Control, John Wiley and Sons, New York, 1977.

16. Pennttinen, J. and Koivo, H. N., "Multivariable Tuning Regulators for Unknown Systems," Automatica, Vol. 16, 1980, 393-398.

17. Ziegler, J. G. and Nichols, N. B., "Optimum Settings for Automatic Controllers, "ASME Transactions 64, 1942, 759-768.

18. Franklin, G. F. and Powell, J. D., Digital Control of Dynamic Systems, Addison-Weseley, Reading, MA, 1980.

19. Carrier System Design Manual, Part 1 Load Estimating, Carrier Air Conditioning Company, Syracuse, New York, 1960.

20. Hurley, C., May, B., Kelly, G., and Borresen, B., "Direct Digital Control of an Air Handler," EMCS Conference, Houston, Texas, November 5-6, 1981.

21. May, W. B., Borresen, B. A., and Hurley, C. W., "Direct Digital Control of a Pneumatically-actuated Air Handling Unit," (to appear in ASHRAE Trans., Vol. 88, Part 2, 1982).

Table 1. Input Data for figure 9

Item	Symbol	Value	Unit
set point	r	18.3	°C
sampling period	T'	20.0	sec
process dead time	d	21.0	sec
operating temperature range	e_R	10.0	°C
scale factor	S_F	20.0	-

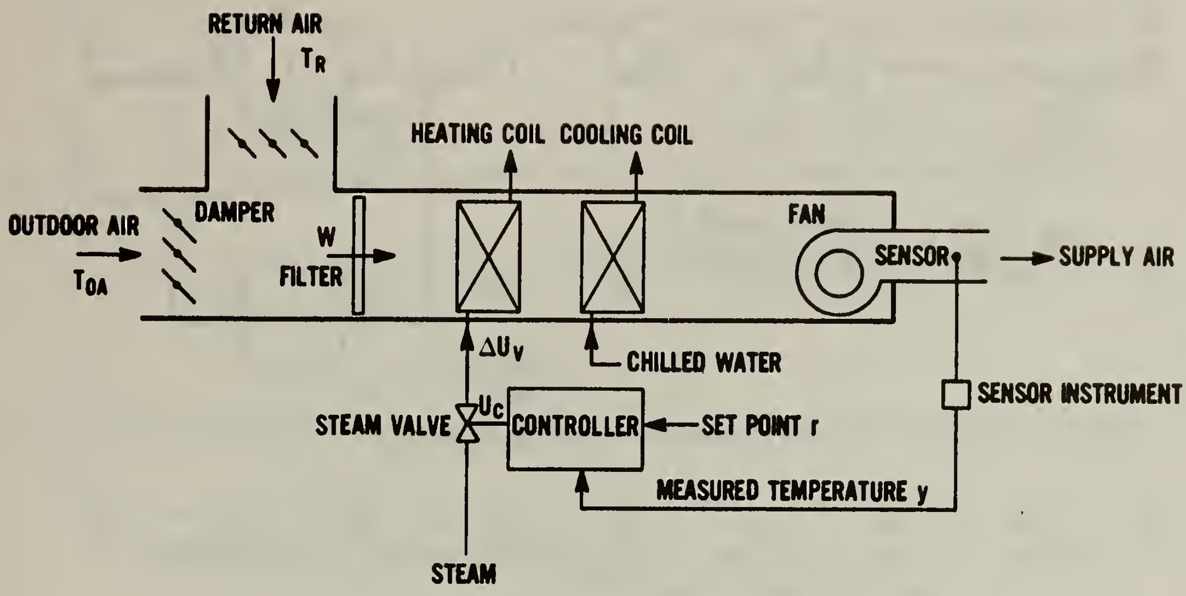

Figure 1. A simple heating, ventilation, and air conditioning (HVAC) system - air handling unit

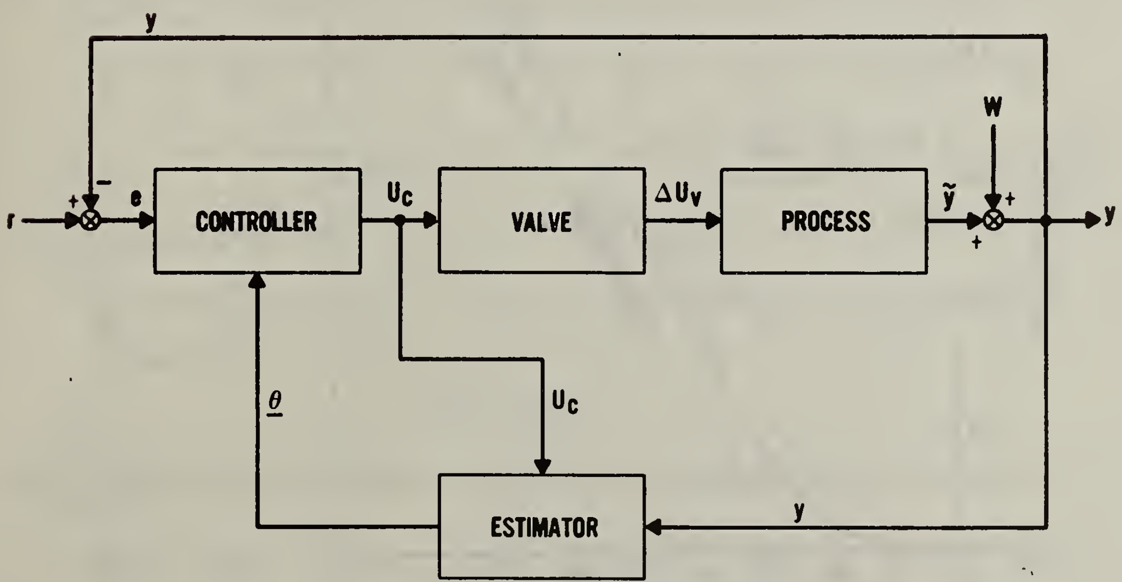

Figure 2. Schematic diagram of the self-tuning controller system

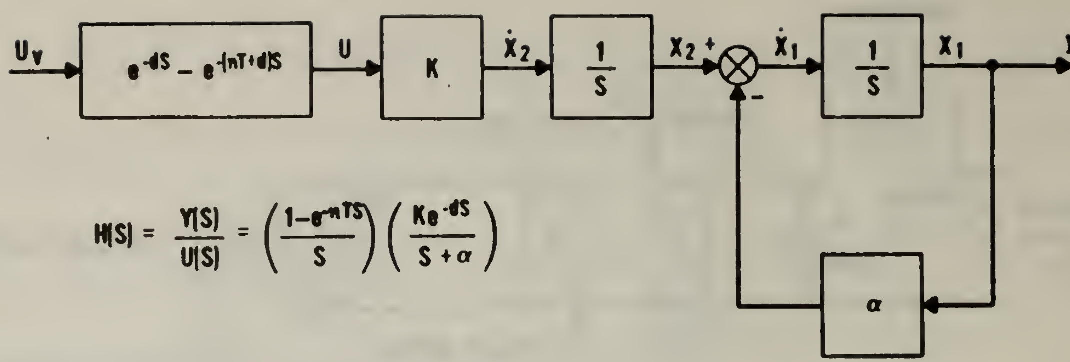

Figure 3. Block diagram of the assumed model of the process to be controlled

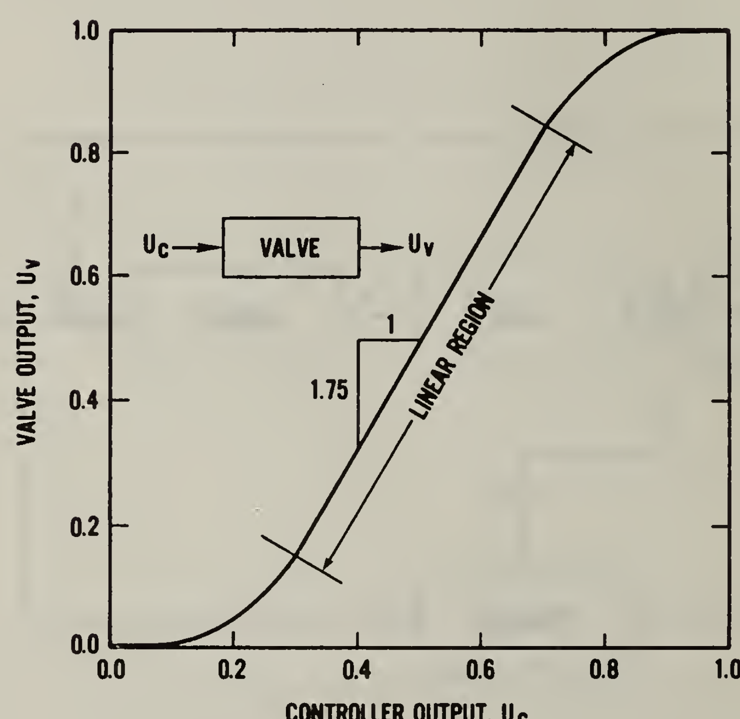

Figure 4. The valve response curve

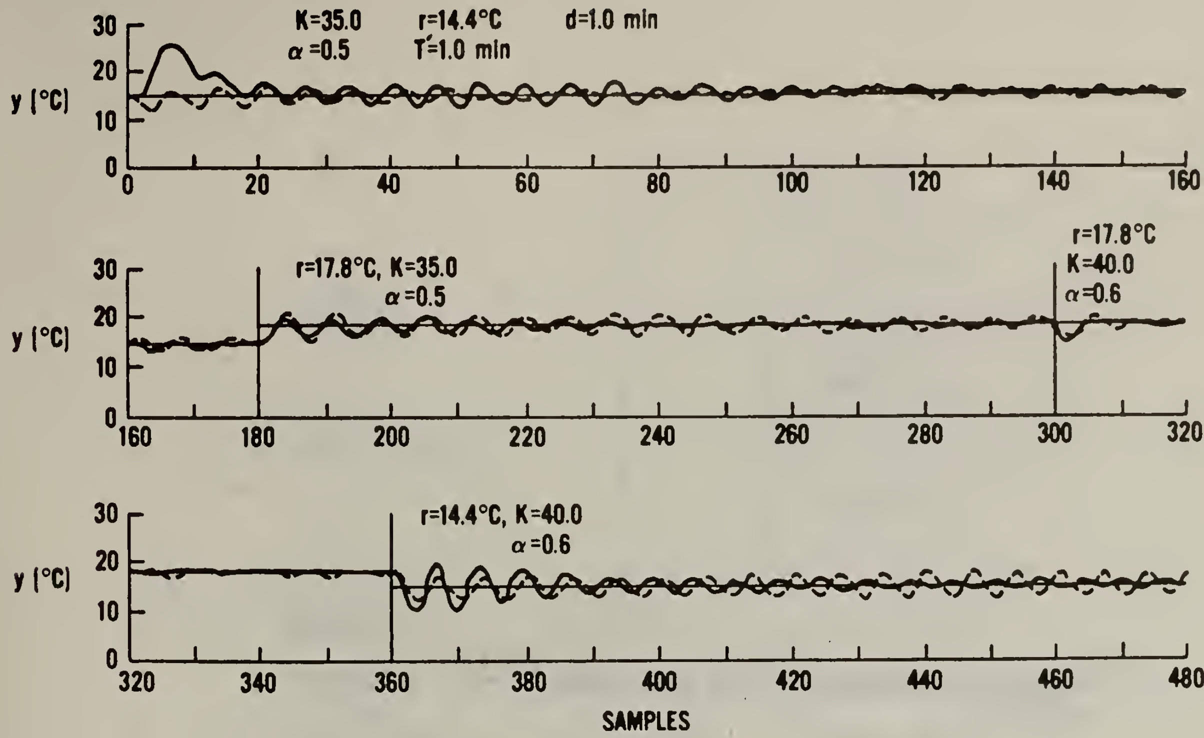

Figure 5. Simulated process output assuming a dead time of 1.0 minute; solid and dotted lines represent adaptive and nonadaptive control, respectively

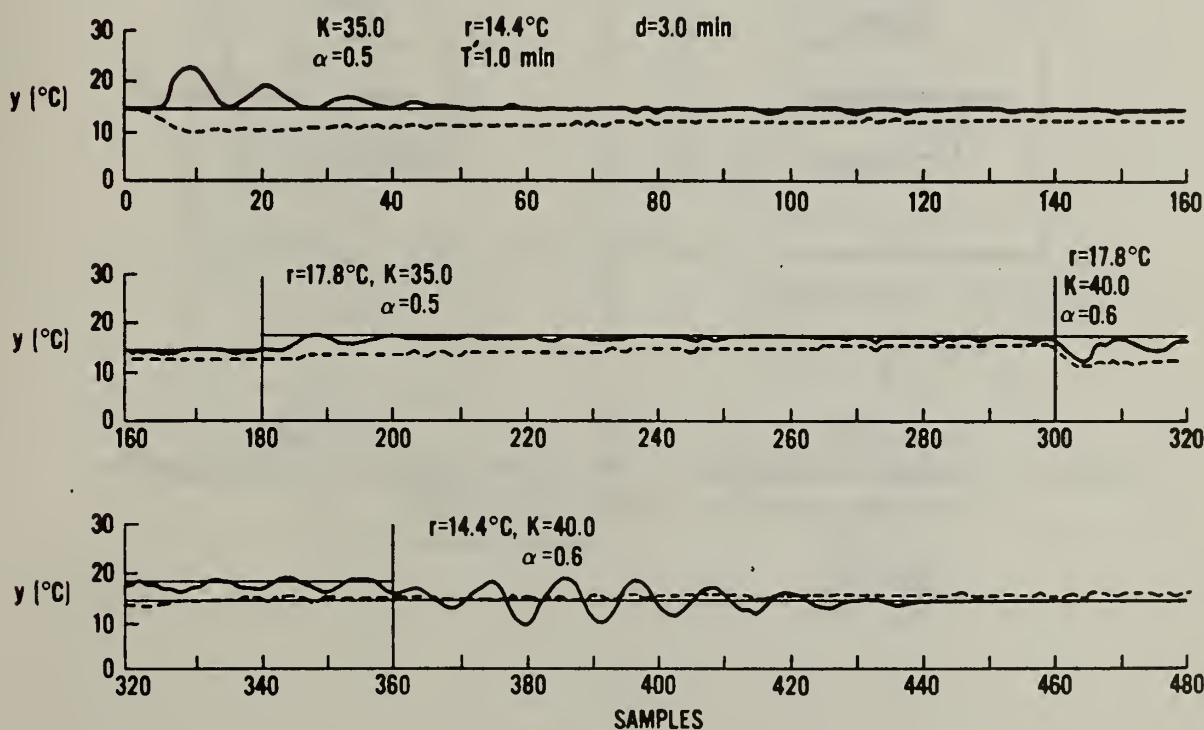

Figure 6. Simulated process output assuming a dead time of 3.0 minutes; solid and dotted lines represent adaptive and nonadaptive control, respectively

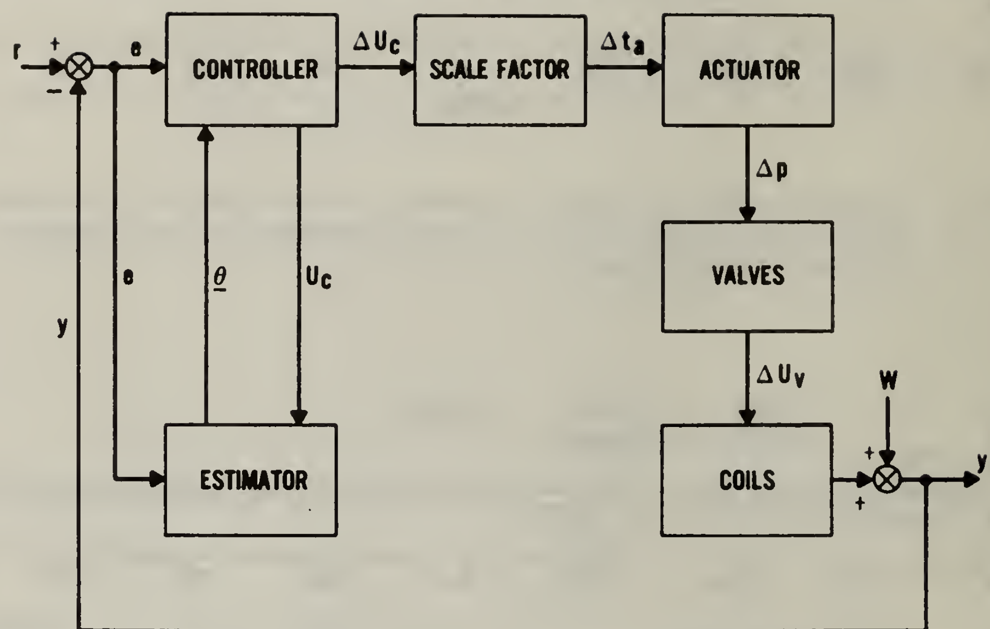

Figure 7. Schematic diagram of the direct digital control system with the adaptive control algorithm

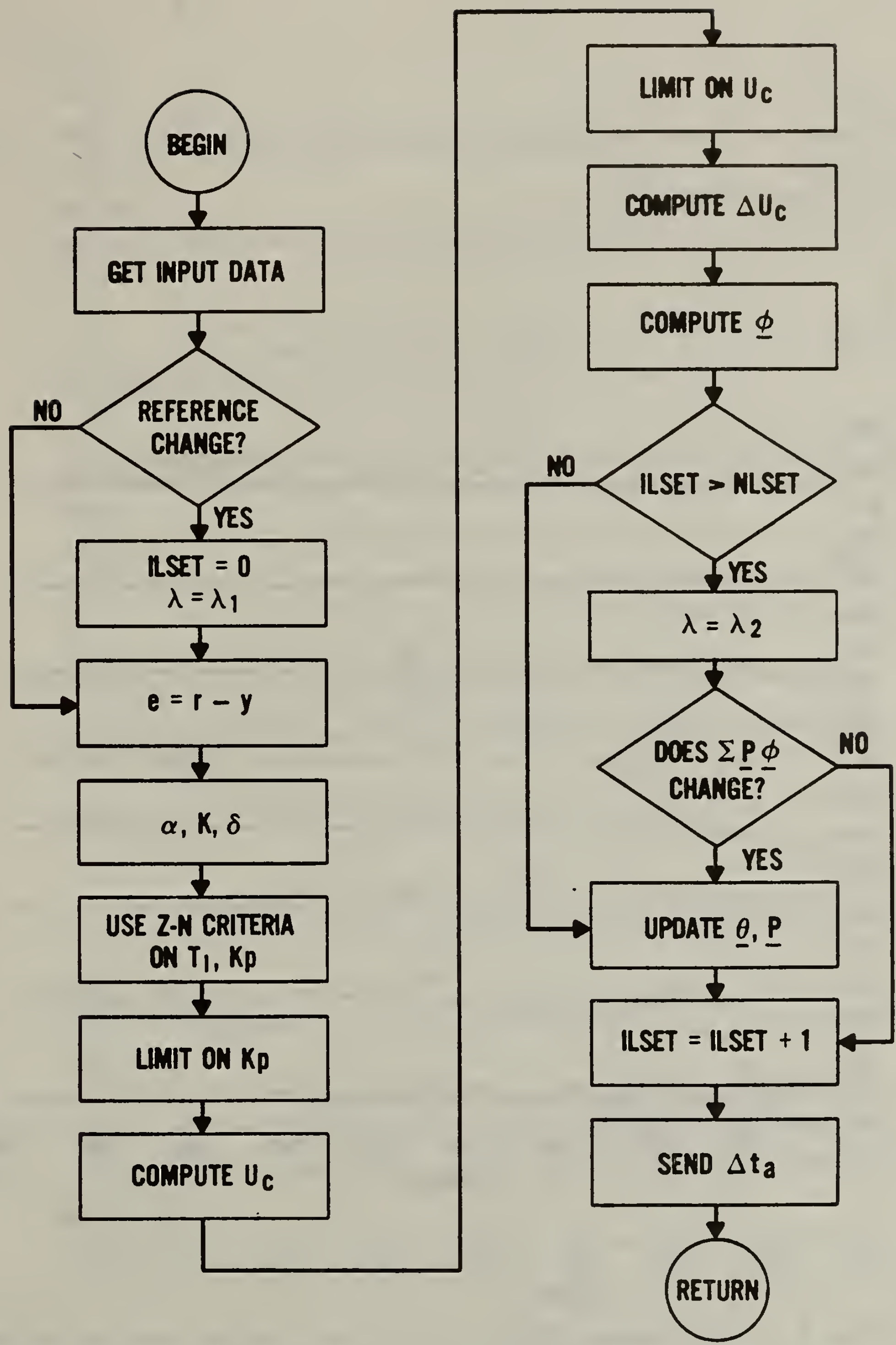

Figure 8. A flow chart of the adaptive algorithm for direct digital control

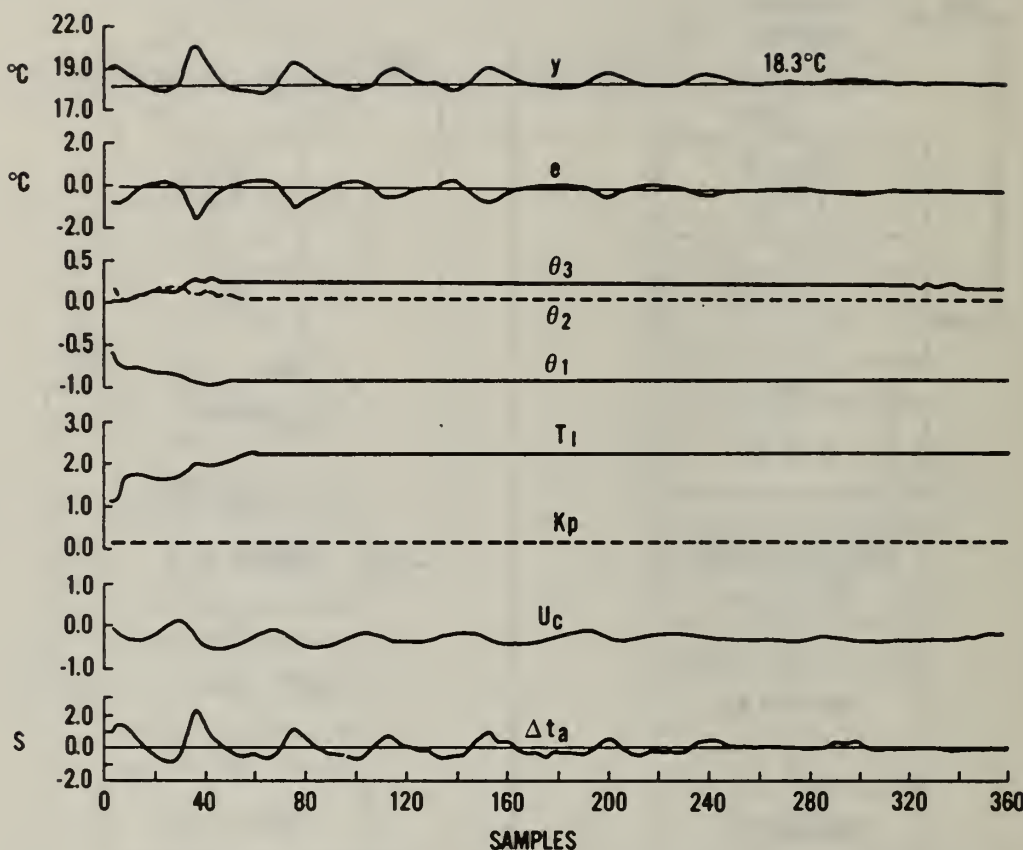

Figure 9. Experimental results of direct digital control at the set point 18.3°C

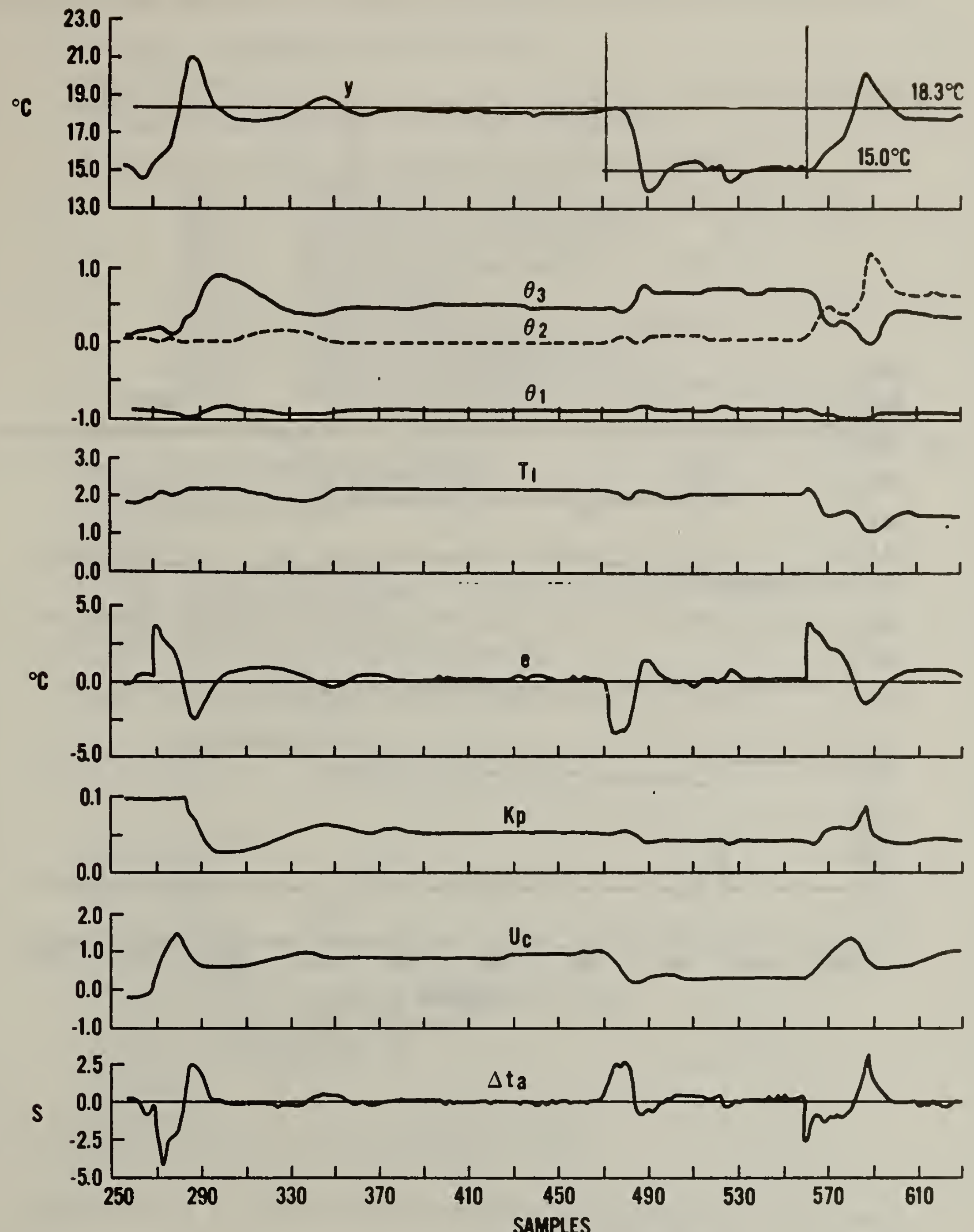

Figure 10. Experimental results of direct digital control with two set points 15.0°C and 18.3°C (typical case)

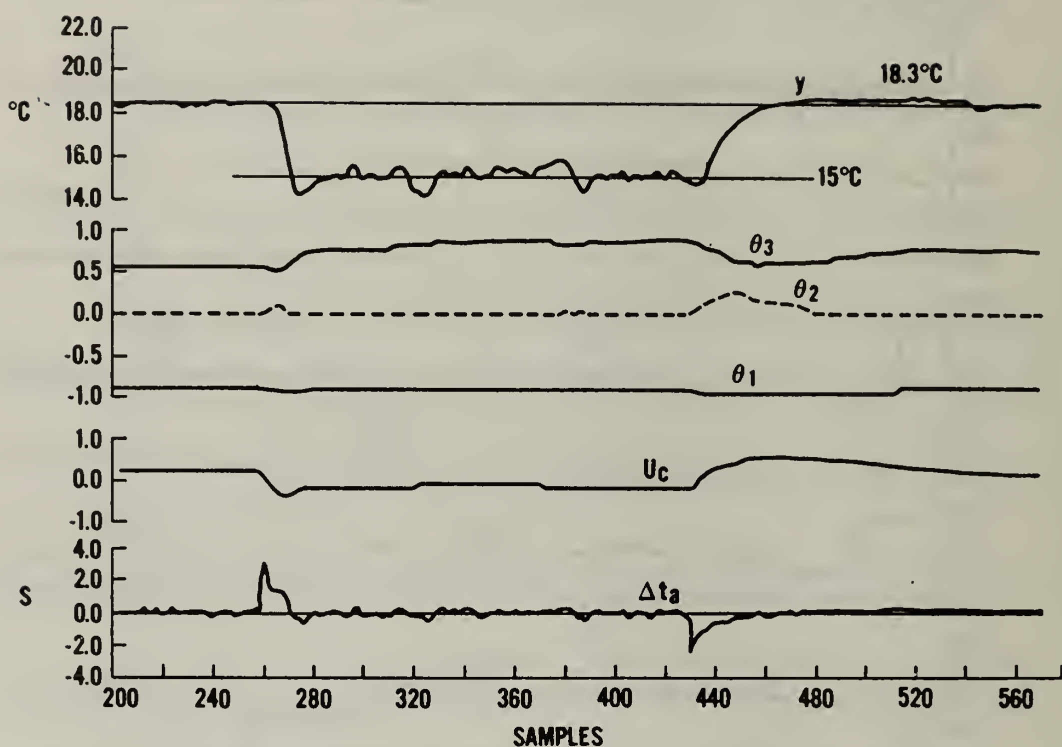

Figure 11, An example of good performance due to satisfactory choice of S_F

APPENDIX

```
C ************************************************************************
C
C         PICNTA:  THE ADAPTIVE PI-CONTROLLER
C
C         VERSION 0628A  C.P.
C         THIS PICNTA SUBROUTINE IS CALLED BY THE MAIN PROGRAM FID09B.FOR
C         AND CALLS THE ADAPTIVE PI-CONTROL ALGORITHM.
C
C         THE FOLLOWING DATA ARE ENTERED VIA FID09B PROGRAM:
C                 ER=      TEMPERATURE RANGE IN C
C                 TDED=    TRANSPORTATION DELAY TIME (DEAD TIME) IN S
C                 SCALE=   SCALE FACTOR ( SYSTEM GAIN)
C                 DTC=     SAMPLING PERIOD IN S
C                 DUMAX=   MAXIMUM ACTUATOR OPERATION TIME IN S
C                 RIN=     INPUT DEVICE
C                 ROUT=    OUTPUT DEVICE
C                 HYSTIM=  COMPENSATION TIME FOR HYSTERISIS IN S
C                 TREF=    SET POINT TEMPERATURE IN C
C                 ANAI=    SUPPLY AIR TEMPERATURE IN C
C                 REST=    LEFTOVER TIME AFTER THE ACTUATOR RESPONDED
C
C ************************************************************************
C
          SUBROUTINE PICNTA(N,DU,E0,REST)
          INTEGER*1 N
          INTEGER DUMAX,RIN,ROUT
          REAL KP
          REAL*8 ANAI
          COMMON /SETPNT/ TREF(16)
          COMMON /REG/ ER(16),TDED(16),SCALE(16),DTC(16),DUMAX(16),
     8    RIN(16),ROUT(16)
          COMMON /ANALOG/ ANAI(16)
          COMMON /BK3/ THETA(3),PHI(3),P(3,3)
C
C         YMAX = HIGHEST PROCESS OUTPUT VALUE IN C
C
          DATA YMAX,IHIT/22.0,0/
C
          CALL DISINT
          J=RIN(N)
C
          TCNT=DTC(N)/60.
          REF=TREF(N)
          YOUT=ANAI(J)
          E0=REF-YOUT
          ERANGE=ER(N)
          TDEAD=TDED(N)/60.
          TACT=SCALE(N)
          DUMAX1=DUMAX(N)
C         TO PREVENT FROM REACHING HI-CUTOFF POINT
          IF(YOUT.GE.YMAX+1.0) GOTO 10
          IF(YOUT.LE.YMAX) GOTO 20
          IF(IHIT.GT.2) GOTO 20
  10      DU=DUMAX1
          IHIT=IHIT+1
          WRITE(5,100)
  100     FORMAT(2X,'*** TOO HIGH TEMP ********')
          GOTO 30
```

```
20      IHIT=0
        CALL ACAPI(TCNT,TDEAD,REF,YOUT,UCNT,DELU,ICOUNT,ERANGE,TIC,KP)
C
C       ACTUATOR OPERATING TIME
C
        DU=-DELU*TACT+REST
C
        WRITE(5,200) ICOUNT,(THETA(I),I=1,3),KP,TIC,UCNT,E0,DU,YOUT
200     FORMAT(I5,3F8.4,3F10.4,3F7.3)
30      CALL ENAINT
C
        RETURN
        END
C ***********************************************************************
C
C       ACAPI: ADAPTIVE CONTROL ALGORITHM FOR PI-CONTROLLER
C
C       TCNT=   SAMPLING PERIOD IN MIN
C       TDED=   DEAD TIME IN MIN -- TDED SHOULD BE GREATER THAN TCNT
C       REF=    SET POINT (REFERENCE VALUE)
C       YOUT=   PROCESS OUTPUT
C       UCNT=   BOUNDED CONTROLLER OUTPUT
C       DELU=   INPUT TO PROCESS
C       ICOUNT= COUNT INDEX
C       ERANGE= PROCESS OPERATING RANGE
C       TIC=    INTEGRAL TIME IN MIN
C       KP=     PROPORTIONAL GAIN
C
C ***********************************************************************
C
        SUBROUTINE ACAPI(TCNT,TDED,REF,YOUT,UCNT,DELU,ICOUNT,ERANGE,
     &  TIC,KP)
        REAL LAMBDA,LAM(2),KP
        DIMENSION UTEMP(20)
        COMMON /BK1/ UCN(20),ERROR,ERLAST,NDEDTC,URANGE
     &         /BK2/ YEST,LAMBDA,PPHSUM,PPHSM1,ILSET,NLSET,EPSPH
     &         /BK3/ THETA(3),PHI(3),P(3,3)
        DATA EPSREF,EPSPH,LAM(1),LAM(2)/0.01,0.01,0.98,1.00/
C       ITMAX=24 HRS*60/(20/60)
        DATA ITCNT,ITINTL,ITMAX/0,30,4320/
C
C       INITIALIZATION
C
        IF(ITCNT.GE.1) GOTO 30
        DO 10 I=1,3
        DO 10 J=1,3
        THETA(I)=0.0
        P(I,J)=0.0
        IF(I.EQ.J) P(I,J)=1.0
10      CONTINUE
        THETA(1)=-0.9
C
        NDEDTC=IFIX(TDED/TCNT)
        NP=NDEDTC+5
        REF1=REF
        NLSET=30
        LAMBDA=LAM(1)
        ERLAST=0.0
```

```
      PPHSM1=1.0
      IF(ABS(YOUT-REF).GT.0.02*ERANGE) YOUT=1.01*REF
      ERROR=REF-YOUT
      URANGE=0.1
      DELU=0.0
      YEST1=0.0
      DO 20 J=1,NP
20    UCN(J)=0.001
      ILSET=0
      GOTO 90
C
C     STEP CHANGE IN SET POINT
C
30    IF(ABS((REF-REF1)/REF1).LE.EPSREF) GOTO 40
      ILSET=0
      LAMBDA=LAM(1)
C
C     CONTROLLER OPERATION
C
40    ERROR=REF-YOUT
      IF(ITCNT.GT.ITINTL) GOTO 60
50    ERROR=-ERROR
      YOUT=YOUT-ERROR*2.0
60    CALL PIZN(TCNT,UCNT,ERANGE,TIC,KP)
      UCNT0=UCNT
C
C     BOUNDING THE CONTROLLER OUTPUTS
C
      ULO=-URANGE
      UHI= URANGE
      IF(UCNT.LT.ULO) UCNT=ULO
      IF(UCNT.GT.UHI) UCNT=UHI
C
C     SHIFTING ONE TIME STEP BACKWARDS
C
      UTEMP(1)=UCNT
      DO 70 J=2,NP
70    UTEMP(J)=UCN(J-1)
      DO 80 J=1,NP
80    UCN(J)=UTEMP(J)
C
C     INPUT SIGNAL TO THE PROCESS
C
      IF(ITCNT.LE.ITINTL) GOTO 90
      DELU=UCN(NDEDTC+1)-UCN(NDEDTC+2)
C
C     DEALING WITH SATURATION CONDITION
C
      IF(UCNT.EQ.ULO.AND.UCNT.EQ.UCN(2)) DELU=UCNT0-UCNT1
      IF(UCNT.EQ.ULO.AND.DELU.GT.0.0) DELU=0.0
      IF(UCNT.EQ.UHI.AND.UCNT.EQ.UCN(2)) DELU=UCNT0-UCNT1
      IF(UCNT.EQ.UHI.AND.DELU.LT.0.0) DELU=0.0
      IF(UCNT.EQ.UCN(NP)) DELU=5.*DELU
C
C     ESTIMATION OF PARAMETERS
C
90    YEST=YOUT-REF
      PHI(1)=-YEST1
```

```
      PHI(2)=UCN(NDEDTC+1)
      PHI(3)=UCN(NDEDTC+2)
      IF(ILSET.GT.NLSET) LAMBDA=LAM(2)
      CALL ESTIM
C
C     CHANGE CURRENT VALUES TO THE LAST
C
      YEST1=YEST
      REF1=REF
      ERLAST=ERROR
      PPHSM1=PPHSUM
      UCNT1=UCNT0
C
      ILSET=ILSET+1
      ITCNT=ITCNT+1
      IF(ITCNT.EQ.ITINTL) ILSET=0
      IF(ITCNT.LE.ITINTL) GOTO 50
C
C     RE-INITIALIZATION EVERY 24 HOURS
C
      ICOUNT=ITCNT-ITINTL
      IF(ICOUNT.EQ.ITMAX+1) ITCNT=0
C
      RETURN
      END
C ************************************************************************
C
C     PIZN: ADAPTIVE PI- CONTROLLER
C
C *************************************************************************
C
      SUBROUTINE PIZN(TCNT,UCNT,ERANGE,TIC,KP)
      REAL KP,KPMIN,KPMAX
      COMMON /BK1/ UCN(20),ERROR,ERLAST,NDEDTC,URANGE
     8       /BK3/ THETA(3),PHI(3),P(3,3)
      DATA CTI/1.1/,KPMAX/0.1/,KPMIN/0.01/
C
C     ADAPTIVE PI-CONTROLLER
C
      ALF=-ALOG(-THETA(1))/TCNT
      GN=ALF*(THETA(2)+THETA(3))/(1.+THETA(1))
      DELT=-ALOG(ALF*THETA(3)/GN-THETA(1))/(ALF*TCNT)
      IF(DELT.GE.0.999) DELT=0.999
C
C     MODIFIED ZIEGLER-NICHOLS CRITERIA
C
      TDED1=(NDEDTC+1-DELT)*TCNT
      TIC=CTI*TDED1
      TIFIX=CTI*(NDEDTC+1)*TCNT
      CKP=0.5*KPMAX
      KP=CKP/(GN*TDED1)
      IF(KP.GE.KPMAX) KP=KPMAX
      IF(KP.LE.KPMIN) KP=KPMIN
      UCNT=UCN(1)+KP*((1.+TCNT/TIC)*ERROR-ERLAST)
      URANGE=KPMAX*(1.+TCNT/TIFIX)*ERANGE
C
      RETURN
      END
```

```
C ***********************************************************************
C
C        ESTIM: ESTIMATION OF PARAMETERS USING A RECURSIVE LEAST SQUARES
C               METHOD
C
C***********************************************************************
C
         SUBROUTINE ESTIM
         REAL LAMBDA
         DIMENSION PPHI(3)
         COMMON /BK2/ YEST,LAMBDA,PPHSUM,PPHSM1,ILSET,NLSET,EPSPH
     8          /BK3/ THETA(3),PHI(3),P(3,3)
C
         DO 10 J=1,3
         PPHI(J)=0.
         DO 10 K=1,3
10       PPHI(J)=PPHI(J)+P(J,K)*PHI(K)
         PPHSUM=0.
         DO 20 J=1,3
20       PPHSUM=PPHSUM+PPHI(J)
C
         E1=0.
         DO 30 J=1,3
30       E1=E1+THETA(J)*PHI(J)
         E=YEST-E1
C
         IF(ILSET.LT.NLSET) GOTO 40
         IF(PPHSM1.NE.0..AND.ABS((PPHSUM-PPHSM1)/PPHSM1).LE.EPSPH) RETURN
C
40       CONTINUE
         DO 50 J=1,3
50       THETA(J)=E*PPHI(J)+THETA(J)
         IF(THETA(1).GE. 0.0) THETA(1)=-0.0001
         IF(THETA(1).LE.-1.0) THETA(1)=-0.999
         IF(THETA(2).LT.0.) THETA(2)= 0.0001
         IF(THETA(3).LT.0.) THETA(3)= 0.0001
C
         R=LAMBDA
         DO 60 J=1,3
60       R=R+PHI(J)*(P(J,1)*PHI(1)+P(J,2)*PHI(2)+P(J,3)*PHI(3))
         R=1./R
C
         DO 70 J=1,3
         DO 70 K=1,3
70       P(J,K)=(P(J,K)-PPHI(J)*R*PPHI(K))/LAMBDA
C
         RETURN
         END
```

NBS-114A (REV. 2-80)

U.S. DEPT. OF COMM. BIBLIOGRAPHIC DATA SHEET (See instructions)	1. PUBLICATION OR REPORT NO. NBSIR-82-2591	2. Performing Organ. Report No.	3. Publication Date November 1982

4. TITLE AND SUBTITLE

ADAPTIVE ALGORITHM FOR THE CONTROL OF A BUILDING AIR HANDLING UNIT

5. AUTHOR(S)

Cheol Park and Alexander J. David

6. PERFORMING ORGANIZATION *(If joint or other than NBS, see instructions)*

NATIONAL BUREAU OF STANDARDS
DEPARTMENT OF COMMERCE
WASHINGTON, D.C. 20234

7. Contract/Grant No.

8. Type of Report & Period Covered

9. SPONSORING ORGANIZATION NAME AND COMPLETE ADDRESS *(Street, City, State, ZIP)*

Office of Building and Community Systems
U.S. Department of Energy
Washington, D.C. 20585

U.S. Navy Civil Engineering Laboratory
U.S. Department of Defense
Washington, D.C.

10. SUPPLEMENTARY NOTES

☐ Document describes a computer program; SF-185, FIPS Software Summary, is attached.

11. ABSTRACT *(A 200-word or less factual summary of most significant information. If document includes a significant bibliography or literature survey, mention it here)*

The use of adaptive control algorithms was studied for microprocessor driven direct digital control of elementary heating and cooling subsystems. An algorithm was designed for digital regulation of a linear, time-invariant first-order system with a system dead time. A recursive least squares algorithm was used to estimate, on-line, the parameters of the time-invariant linear system. The parameter estimates were then used to calculate the feedback gains of a Proportional plus Integral (PI) controller.

Through computer simulations, the adaptive-parameter PI-controller was compared with a constant-parameter PI-controller. On the basis of favorable simulation results, the adaptive algorithm was implemented for direct digital control of an air handling unit in a laboratory building at the National Bureau of Standards, Gaithersburg, Maryland. The convergence of the parameter estimates and the step response proved to be satisfactory provided the system was operating in a linear or weakly non-linear region, and was in steady or quasi-steady state. By selecting a proper scale factor, improved performance may be obtained when system characteristics vary.

12. KEY WORDS *(Six to twelve entries; alphabetical order; capitalize only proper names; and separate key words by semicolons)*

adaptive control; air handling unit; direct digital control; energy management and control systems; HVAC system control; parameter estimator; PI-controller; recursive least squares algorithm; self-tuning control algorithm.

13. AVAILABILITY

[X] Unlimited
☐ For Official Distribution. Do Not Release to NTIS
☐ Order From Superintendent of Documents, U.S. Government Printing Office, Washington, D.C. 20402.
[X] Order From National Technical Information Service (NTIS), Springfield, VA. 22161

14. NO. OF PRINTED PAGES

48

15. Price

$7.50